

# Claims

- [c1] 1. A method for executing a database statement, the method comprising:  
preparing at least one template for execution of a statement against a database;  
storing said at least one template in a shared cache available to a plurality of database connections;  
in response to a request to execute a particular statement on a given database connection, determining whether a template for said particular statement is available in the shared cache;  
if the template is available in the shared cache, creating a database statement based on the template for execution on the given database connection; and  
executing the database statement on the given database connection.
- [c2] 2. The method of claim 1, wherein said preparing step comprises preparing a structure for execution of the statement against a database.
- [c3] 3. The method of claim 2, wherein said structure comprises an executable structure for executing the statement on a database connection.

- [c4] 4. The method of claim 1, wherein the request to execute a particular statement on a given database connection is received at an application server.
- [c5] 5. The method of claim 1, wherein the request to execute a particular statement on a given database connection is received at a database client.
- [c6] 6. The method of claim 1, wherein said shared cache is available at a database client to application threads accessing a remote database server.
- [c7] 7. The method of claim 6, wherein said database client comprises an application server.
- [c8] 8. The method of claim 1, wherein said given database connection includes a connection allocated from a connection pool.
- [c9] 9. The method of claim 1, further comprising:  
if the template is unavailable, preparing a template for execution of the particular statement against the database;  
placing the template in the shared cache; and  
creating a database statement based on the template for execution on the given connection.
- [c10] 10. The method of claim 1, wherein said storing step in-

cludes making said shared cache available on a first database connection.

- [c11] 11. The method of claim 10, wherein a template in said shared cache is cloned for use on a second database connection.
- [c12] 12. The method of claim 1, further comprising:  
assigning a unique identifier to each statement to be executed against the database.
- [c13] 13. The method of claim 12, wherein said determining step includes using said unique identifier to determine whether a template is available in the shared cache.
- [c14] 14. The method of claim 12, wherein said shared cache comprises an array of templates indexed based on said unique identifier.
- [c15] 15. The method of claim 1, further comprising:  
synchronizing access to the shared cache from application threads on a plurality of database connections.
- [c16] 16. The method of claim 1, wherein said creating step includes reusing immutable portions of a template.
- [c17] 17. The method of claim 1, wherein said creating step includes duplicating mutable portions of a template.

- [c18] 18. The method of claim 17, wherein said mutable portions include nodes of the template having mutable children.
- [c19] 19. The method of claim 17, wherein said mutable portions include nodes of the template having mutable fields.
- [c20] 20. The method of claim 1, wherein said creating step includes attaching the database statement to the given database connection.
- [c21] 21. The method of claim 1, wherein said executing step includes returning results of executing the database statement.
- [c22] 22. A computer-readable medium having processor-executable instructions for performing the method of claim 1.
- [c23] 23. A downloadable set of processor-executable instructions for performing the method of claim 1.
- [c24] 24. A system for executing a database statement, the system comprising:
  - at least one template comprising an executable structure for execution of a statement against a database;
  - a shared cache for storing said at least one template at-

tached to a first connection;  
a locator module for locating a template corresponding to a particular statement in the shared cache in response to a request to execute the particular statement on a second database connection;  
a cloning module for cloning the template to create a database statement for execution on the second database connection; and  
an execution module for executing the database statement on the second database connection.

- [c25] 25. The system of claim 24, wherein said at least one template comprises at least one database prepared statement.
- [c26] 26. The system of claim 24, wherein said at least one template comprises at least one prepared statement in executable form for execution against a database.
- [c27] 27. The system of claim 24, wherein the request to execute a particular statement on a second database connection is received at an application server.
- [c28] 28. The system of claim 24, wherein the request to execute a particular statement on a second database connection is received at a database client.
- [c29] 29. The system of claim 24, wherein said shared cache is

available at a database client to application threads accessing a remote database server.

[c30] 30. The system of claim 29, wherein said database client comprises an application server.

[c31] 31. The system of claim 24, wherein said second database connection includes a connection allocated from a connection pool.

[c32] 32. The system of claim 24, further comprising:  
a module for preparing a template for execution of the particular statement and placing the template in the shared cache if the locator module determines that the template is unavailable.

[c33] 33. The system of claim 24, further comprising:  
a module for assigning a unique identifier to a statement to be executed against the database.

[c34] 34. The system of claim 33, wherein the locator module uses said unique identifier to determine whether a template is available in the shared cache.

[c35] 35. The system of claim 33, wherein said shared cache comprises an array of templates indexed based on said unique identifier.

[c36] 36. The system of claim 24, further comprising:

a module for synchronizing access to the shared cache from application threads on a plurality of database connections.

- [c37] 37. The system of claim 24, wherein the cloning module reuses immutable portions of a template in cloning the template.
- [c38] 38. The system of claim 24, wherein the cloning module duplicates mutable portions of a template in cloning the template.
- [c39] 39. The system of claim 38, wherein said mutable portions of a template include nodes of the template having mutable children.
- [c40] 40. The system of claim 38, wherein said mutable portions of a template include nodes of the template having mutable fields.
- [c41] 41. The system of claim 24, wherein the cloning module attaches the database statement to the second database connection.
- [c42] 42. The system of claim 24, wherein the execution module returns any results of executing the database statement.
- [c43] 43. A method for executing a prepared statement

against a database, the method comprising:  
generating a prepared statement comprising an executable structure for a particular statement to be executed against the database;  
attaching said prepared statement to a first connection to the database;  
in response to a request to execute the particular statement on a second connection to the database, cloning the prepared statement for execution on said second connection; and  
executing the prepared statement against the database on said second connection.

[c44] 44. The method of claim 43, wherein said generating step includes generating a plurality of prepared statements for executing a plurality of statements against the database.

[c45] 45. The method of claim 43, wherein said prepared statement comprises a template for executing the particular statement against the database on a database connection.

[c46] 46. The method of claim 43, wherein the request to execute the particular statement on a second connection is received at an application server.



- [c47] 47. The method of claim 43, wherein the request to execute the particular statement on a second connection is received at a database client.
- [c48] 48. The method of claim 43, wherein said prepared statement is stored in a cache at a database client.
- [c49] 49. The method of claim 48, wherein said database client comprises an application server.
- [c50] 50. The method of claim 43, wherein said second connection includes a database connection allocated from a connection pool.
- [c51] 51. The method of claim 43, wherein said generating step further comprises assigning a unique identifier to the particular statement.
- [c52] 52. The method of claim 51, wherein said cloning step includes using said unique identifier to locate the prepared statement.
- [c53] 53. The method of claim 51, wherein the prepared statement is stored in a shared cache indexed based on said unique identifier.
- [c54] 54. The method of claim 43, further comprising:  
synchronizing access to the prepared statement from application threads on a plurality of database connec-

tions.

- [c55] 55. The method of claim 43, wherein said cloning step includes reusing immutable portions of a template.
- [c56] 56. The method of claim 43, wherein said cloning step includes duplicating mutable portions of a template.
- [c57] 57. The method of claim 56, wherein said mutable portions include nodes of the template having mutable children.
- [c58] 58. The method of claim 56, wherein said mutable portions include nodes of the template having mutable fields.
- [c59] 59. The method of claim 57, wherein said cloning step includes attaching the database statement to said second connection.
- [c60] 60. The method of claim 43, wherein said executing step includes returning results.
- [c61] 61. A computer-readable medium having processor-executable instructions for performing the method of claim 43.
- [c62] 62. A downloadable set of processor-executable instructions for performing the method of claim 43.

